

## [補足1] GA-ANNを用いた倒立振り子制御

GA-ANNの理解を深めるため、以下にGA-ANNを用いた簡単な制御器の例を示す。

今、3層のANNを考える。ANNの入力変数は、振り子の角度 $\theta$ と角速度 $\dot{\theta}$ であり、出力変数は台車に加える力である。入力-隠れ層重み $V$ 、隠れ-出力層重み $W$ はGAにより進化的に決定する。

( $V, W$ の値によって台車に加える力 $F$ が $-k \sim k$ まで変化する。)

GAの適合度関数(振り子の安定評価)は演習資料で示すように様々な関数が考えられる。参考プログラムでは、シミュレーションを開始してからsimulationTimeステップ目からtestTimeステップ経過後に計測される最大振幅値を適合度として定義した。

参考プログラムでは以前作成したBPNNクラスを継承したGANNクラスを用いた。

GANNクラスでは親クラスのBPNNのネットワーク構造・入出力計算(InputToOutput)を利用し、NNの重みをGAで進化的に獲得するクラスとなる。

## ■ オブジェクトの定義

```
GANN gann = new GANN(入力層ユニット数, 隠れ層ユニット数, 出力層ユニット数);
```

## ■ 個体数・突然変異率の設定

```
gann.SetGAParameter(個体数, 突然変異率);
```

## ■ 適合度計算

```
gann.CalcFitnessFunction(シミュレーション時間, 安定検証時間, 初期角度);
```

## ■ 適合ごとに遺伝子のソート(fitness[0]が最良適合度)+淘汰

```
gann.Sort();
```

## ■ 突然変異

```
gann.Mutation();
```

## ■ 交叉

```
gann.Crossover();
```

## ■ 入力に応じた力の計算 in:double []

```
f=gann.CalcForce(in);
```

サンプルプログラムでは、以下の初期条件となっている。

## ■ 初期値

角度 $\theta = 2\pi/3$ , 各速度 $\dot{\theta} = 0$

## ■ NN及びGA

入力数:2, 隠れ層:10, 出力数:1

個体数:10, 染色体の長さ:ANNの重みの個数分, 突然変異率:40%

## ■ シミュレーション

各世代で獲得した重みで5000Stepシミュレーションを実行し、その後20000Step間の最大振幅値を適合度とした。

このプログラムを実行すると、 $2\pi/3$  の角度から振り子が持ち上げる制御を獲得する様子がみられるが、この時、適合度関数や入力変数に台車の位置・速度情報が入っていないため、棒が直立したにもかかわらずどちらか一方に台車が移動するといった現象が見られ不十分である。このプログラムを参考にして各自 GANN を用いた倒立振り子のプログラムを完成させてください。(異なる適合度関数や入力情報などの振る舞いはどうか?)

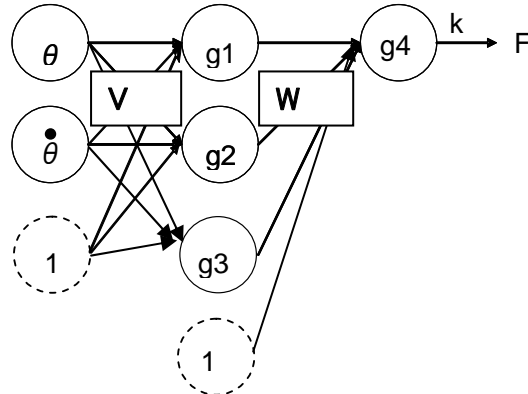
ANN :  $f = g_3(\theta, \dot{\theta}, w, k)$

特性関数 :  $g_{1,2,3}(u) = \frac{1}{1 + e^{-u}}$ ,

$g_4(u) = \frac{2}{1 + e^{-u}} - 1$

制御器 :  $F = k \cdot g_4$

$k$  : ゲイン



\*ここで用いる出力部の特性関数は[-1,+1]の範囲をとるものとしている。以下のような、式を用いると良い。

